UNSUPERVISED MODELING OF USER ACTIONS IN A DIALOG CORPUS

Donghyeon Lee, Minwoo Jeong, Kyungduk Kim, Gary Geunbae Lee

Department of Computer Science and Engineering
Pohang University of Science & Technology, Pohang, South Korea
{semko, stardust, getta, gblee}@postech.ac.kr

ABSTRACT

In data-driven spoken dialog system development, developers should prepare a dialog corpus with semantic annotation. However, the labeling process is a laborious and time consuming task. To reduce human efforts, we propose an unsupervised approach based on non-parametric Bayesian Hidden Markov Model to the problem of modeling user actions. With the non-parametric model, system designers do not need to determine the number and type of user actions. In the experiments, we evaluated the clustering results by comparing them to the human annotation. We also tested a dialog system that used models trained from the automatically annotated corpus with a user simulation.

Index Terms**—** Dialog System, unsupervised learning

1. INTRODUCTION

A dialog corpus is an essential resource for developing a data-driven dialog system [1][2] that consists of three major components: automatic speech recognition (ASR), spoken language understanding (SLU) and dialog management (DM). In most cases, a dialog corpus is acquired by means of the Wizard-of-Oz technique. Sometimes, a hand-crafted human-human dialog corpus is collected for the initial development. To build models for dialog systems, the corpus should be labeled with a semantic annotation which contains a user action (UA), named entities (NE), and system actions (SA). The annotation process requires tedious human effort.

To reduce human efforts for the UA annotation, Tur et al. [3] describe active and semi-supervised learning methods for SLU. Recently, as on-line conversation and social networking grow, so does the need for the research on dialog act (DA) recognition [4]-[6]. Our proposing method is related to these studies introduced so far, but distinguished by targeting a dialog corpus. Although, both UA and DA represent user-intention they are distinguished from each other by its domain dependency, i.e., the UA is domain-specific whereas the DA is not.

The limitation of the unsupervised DA tagging in previous studies is that the number of DA types is fixed by the model. However, it is difficult to find the adequate number of DA types without human analysis. In [5], the training was repeated on some varying numbers of DA types. Although Crook et al. [7] used this approach in a travel-planning dialog corpus, the sequential structure of dialog was not reflected in their work.

We address the UA labeling problem in an unsupervised manner by using the hierarchical Dirichlet process Hidden Markov Model (HDP-HMM) [8][9]. Our model not only has a flexible number of UA types but also considers dialog structure. Training process includes learning the number of UA types. In the previous studies, [5]-[7] only evaluated the performance of the clustering result. In addition to this, we present performance evaluation of a dialog system to show the effectiveness on an end-use application.

The remainder of this paper is structured as follows. Section 2 describes the task, section 3 describes our method for unsupervised modeling of UAs, section 4 provides an evaluation of the proposed method, and Section 5 concludes with a discussion of future work.

2. PROBLEM DEFINITION

Our goal is to group user utterances in a dialog corpus into UA types. We assume the NEs are predetermined before the UA annotation. Assuming predefined label is reasonable because the NE often labeled through a separated process.

Fig. 1 shows an example dialog in our dialog corpus, which consists of a sequence of user and system utterances. The small Gothic characters (Fig. 1) indicate NE types. The system utterances are template forms not complete sentences. The rightmost column represents an UA or an SA corresponding to the utterance. Specifically, the boldfaced labels mean the UAs that are targeted in our method. Although we have the UA labels in the corpus, they are not used for model learning, but for references in our experiments.

In traditional approaches, for the human annotation, developers should define UA types whereas it is not necessary in our approach. After clustering is finished, the UA is filled with the cluster ID for each user utterance in the dialog corpus. Although the cluster ID is not in human-readable format, it is sufficient for spoken dialog system development because the system is able to decide whether two arbitrary utterances have the same UA type or not. The cluster ID can be renamed with little human effort when a meaningful label is needed.

User	*엄영일*$_{PER\_NAME}$ *소장*$_{PER\_TITLE}$ 님 방 을 찾 고 있 습니다 . *I'm looking for Manager Um Young-il's room.*	**search_per_loc**
System	<per_name> <per_title> 의 오피스는 <room_num>입니다. *<per_name> <per_title> office is <room_num>.*	inform_P(RoomNumber)
User	*201호*$_{ROOM\_NUM}$ 는 어디 입니까 ? *Where is No. 201?*	**search_loc**
System	<loc_room_name>는 <loc_losition>에 있습니다. *<loc_room_name> is located in <loc_position>.*	inform_B(Position)
User	*201호*$_{ROOM\_NUM}$ 로 안내 부탁 드립니 다 . *Please guide me to No. 201.*	**guide_loc**
System	네 그럼 안내 시작하겠습니다. *Okay, Let's start.*	guide

Fig. 1. *An example dialog in our dialog corpus*

3. UNSUPERVISED MODELING OF USER ACTIONS

Our method is inspired by the model proposed in [6]. We adopted the model on the dialog corpus instead of Twitter. Our method is distinguished from [6] by using the HDP-HMM which is able to learn the number of UAs. In this section, we describe our method for unsupervised modeling of user actions.

3.1. Background

Bayesian HMMs have been applied in unsupervised training [10]. This model is a parametric approach that requires prior estimation of the number of clusters K. However, the HDP-HMM model is a flexible, nonparametric model which allows state spaces of unknown size to be learned from data. This approach defines an a priori distribution on transition matrices over countably infinite state spaces.

3.2. Proposed Model

As in [5], our model structure is based on the content model proposed in [11] for summarization tasks. The content model uses HMM models for topic transitions, with each topic generating a message. We apply this model to the dialog corpus for creating the model. We consider actions as the hidden states, and assume each utterance (sentence) is generated by an action.

Our model basically combines the HDP-HMM with the content model for a dialog corpus. The reason to use this non-parametric approach is that it is not always possible to know the number of UA types in advance. In the existing systems, a human analysis was required to determine the number. It should be automatically set for a fully unsupervised UA labeling.

A graphical representation of our model is shown in Fig. 2. Each dialog D is a sequence of actions z, and each action generates a sentence, represented by a bag of words and entities shown using the V and N plate respectively. The actions z can be of two possible types: the UA and the SA. The cluster space for UAs and SAs is separated during a learning process. That means UAs cannot share clusters with SAs. The state transitions are generated by Multi(π_k) whose prior π_k. The transition prior π_k is generated by a Dirichlet Process with a hyperparameter α' and a base distribution $\boldsymbol{\beta}$. The base distribution $\boldsymbol{\beta}$ is generated by a GEM distribution with a hyperparameter α. Emissions are generated by Multi(ϕ_k) with a prior ϕ_k generated by Dir(ϕ_0) with a symmetric hyperparameter ϕ_0. We also include the two following Bayesian extensions to improve the model as described in next subsections. The state transitions are generated by Multi(π_k)

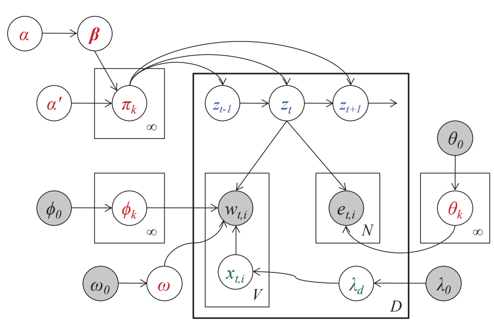

Fig. 2. *A graphical representation of our model*

3.2.1. Word/Entity Model

The user utterance consists of words. Each word belongs to one of two categories: an entity word and a non-entity word. We can easily determine the category for each word from the named-entity annotation in the given corpus. We use the category to improve the model. We divide words that are generated from an action into two categories, shown in Fig. 2. The entity word is replaced with the entity class name. In Fig. 2, the non-entity words and the entities are represented by a bag of words and a bag of entities shown using the V and N plate, respectively. In addition, the word/entity model includes the emission parameter θ_k that is related to the entity. Entities are generated by Multi(θ_k), with a prior θ_k, generated by Dir (θ_0) with a symmetric hyperparameter θ_0. We build the word/entity model to prevent the utterances from being grouped into clusters by a content word that is labeled as an entity. Our goal is to cluster sentences describing the same UA, rather than the same content.

3.2.2. Background Model

We use a unigram language model (LM) as the emission distribution. However, such a model can be distorted by general words that can occur in many utterances and are usually less discriminative among actions. To resolve this problem, we use a background LM for general words. This approach is similar to a Latent Dirichlet Allocation style topic model [12]. Each non-entity word is generated from one of two sources: the current UA and general words. A new hidden variable x determines the source of each word and is drawn from Bern(λ_d) (a Bernoulli distribution) with a parameter λ_d generated by Beta(λ_0) (a beta distribution) with a parameter λ_0. If a word source is the general words, then a word is generated by Multi(ω) with a prior ω generated by Dir(ω_0) with a symmetric hyperparameter ω_0.

4. RESULTS

In our model, the hyperparameters were manually set to $\alpha = 2$, $\alpha' = 5$, $\pi_0 = 0.1$, $\phi_0 = 0.1$, $\omega_0 = 0.1$ and $\lambda_0 = 0.1$. These values are temporary (not optimal). To perform inference, we used Gibbs sampling [13], a stochastic procedure that produces samples from the posterior distributions. The clustering results were obtained after 1,000 iterations. We trained the model 100 times to evaluate and analyze the clustering results.

4.1. Dialog Corpus

For experiments, we used the dialog corpus for an intelligent robot to provide information about a building (e.g., room number, room name, and room type) and person (e.g., name, phone number, and e-mail address). The dialog corpus consists of 1,765 user utterances from 429 dialogs. The average length (user and system turns) of the dialog is 8.23, and the vocabulary size is 267. For the human annotation, we defined 15 UA 10 NE types, and 16 SA types.

4.2. Clustering Evaluation

To accomplish clustering evaluation, we evaluated the clustering performance by purity (Pur), rand index (RI), V-Measure (V-M) and F-Measure (F-M) [14] [15]. Table 1 shows the clustering results when different models were used. It includes the various measures and the number of actual clusters (#C) which are observed. Each number is a mean value with a variance for each measure. In order to compare with other methods, we considered K-means, HDP [7] and HMM [5].

The K-Means and the HMM clustering method require the number of DA types. The number defined in human annotation was given for the models. This causes that the K-Means and the HMM has an advantage than the HDP and the HDP-HMM. The HMM and the HDP-HMM perform better compared to the K-means and the HDP respectively because a dialog structure is reflected.

We introduced our model with two Bayesian extensions in section 3; the word/entity (E) and background (B) model. To evaluate these components, we applied these extensions separately. Table 1 shows that the F-measure score is increased by adding the word/entity model. However, the result was not satisfactory on other measures. The algorithm was not sufficient to solve the problem caused by clusters that are created due to content words. Adding the background model makes a significant improvement on all measures by separating general words that cannot be helpful to clustering DAs. The final model, which contains all of the extensions, achieved the best F-measure score.

The SA can be directly labeled in the corpus acquisition process when a system specification is given. In this experiment, the hidden states for the SAs are fixed because they are already determined in the given corpus. Sometimes, however, a raw dialog corpus has only system utterances without SAs. The last row of table 1 shows performance on that occasion. The performance decrease is unavoidable because SAs also should be clustered.

Confusion matrix for a clustering example is shown in Fig. 3. The example is generated in the final model and shows the best V-measure score. The cluster ID is determined in the learning process. The IDs and labels of the human-labeled DA are represented in Fig. 3. In most cases, utterances labeled as *bye* or *thank you* are clustered together. This collection is beneficial because the utterances seem to have a similar purpose and appear in almost the same situation. Utterances labeled as *search_per_cellphone*, *search_per_mail* or *search_per_phone* are sometimes clustered together. Although requested information is slightly different, their goal is to get information to contact a person. This set is acceptable. On the other hand, the utterances labeled *search_loc* by human annotators as are divided into many clusters. The reason for this division is that the utterance patterns are different according to entities appearing in the utterance.

Models	Pur	RI	V-M	F-M	#C
K-Means	0.6436 (0.0010)	0.7549 (0.0010)	0.5188 (0.0007)	0.3863 (0.0008)	15.00 (-)
HDP	0.5793 (0.0008)	0.7046 (0.0012)	0.4862 (0.0006)	0.4090 (0.0019)	10.62 (2.08)
HMM	**0.7434** (0.0007)	0.7931 (0.0009)	0.6565 (0.0008)	0.4920 (0.0026)	15.00 (-)
HDP-HMM	0.6838 (0.0010)	0.7904 (0.0020)	0.6363 (0.0008)	0.5398 (0.0055)	11.57 (2.31)
+E	0.6816 (0.0012)	0.7848 (0.0022)	0.6319 (0.0008)	0.5400 (0.0057)	11.52 (2.43)
+B	0.7376 (0.0011)	0.7971 (0.0016)	**0.6882** (0.0009)	0.5875 (0.0040)	10.86 (1.84)
+E+B (final)	0.7350 (0.0015)	**0.8092** (0.0016)	0.6877 (0.0010)	**0.5904** (0.0045)	10.70 (2.03)
+E+B-SAs	0.6098 (0.0011)	0.7505 (0.0039)	0.5719 (0.0055)	0.5400 (0.0040)	7.23 (1.31)

Table. 1. *Clustering results for models*

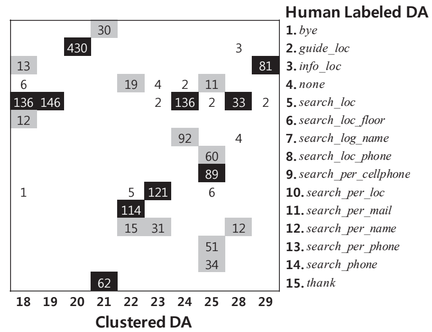

Fig. 3. *Confusion matrix for the clustering example*

4.3. Dialog System Evaluation

In the previous evaluation, we measured the clustering performance by using the manually annotated UA sets as the target clustering. This strategy is not the best way to evaluate the clustering results in two reasons. First, we cannot guarantee that the human annotation is the best answer. Determining the UA types and labeling the UA are ambiguous tasks for humans. Second, the ultimate goal is to use the automatically labeled dialog corpus for a dialog system. Thus, it is necessary to evaluate performance using the dialog system for better clustering evaluation.

In our experiments, we used an EBDM method, which is one of the data-driven dialog modeling techniques [2]. As in many dialog systems, understanding the user action is important to decide the system action in EBDM. Therefore, the UA annotation affects the dialog system's performance.

An agenda graph is a graph of the knowledge sources for a dialog management system to reflect a desired discourse structure. The EBDM framework can use the agenda graph as prior knowledge [16]. We can directly create the agenda graph from our model by using the state transition probabilities. Fig. 4 shows an example agenda graph. The boldfaced labels mean a human labeled DA which has the largest counts in the cluster. We omitted some edges with low weight for simplicity in the graph.

We evaluated the two dialog systems with agenda graphs. The models for the dialog systems are trained on the human-annotated corpus (HC) or the automatically annotated corpus (AC) from the final model. We measured task completion rates (TCR), average dialog lengths (#AvgLen), and the rewards score (SCORE) under various word error rate (WER) conditions. The reward score is similar to the reward function of reinforcement learning used in [18]. For each dialog, the system gets 20 points if the dialog is successfully completed, and loses one point for each dialogue turn. To evaluate the dialog system, we used a dialog simulator proposed in [17]. We used 1000 simulated dialogs and 10-best recognition hypotheses for an automatic evaluation.

Fig. 5 shows the experimental results of the two systems. The system with the HC outperforms the system with the AC, but with high WER, the AC shows slightly better TCR. Although the clustering results and the V-Measure were not satisfactory in the clustering evaluation, the results are encouraging in the dialog evaluation; despite the system with the AC has a handicap because the simulation models are learned from the HC. Moreover, the system with the AC requires considerably less human effort than the system with the HC. As a result, our unsupervised approach is sufficient to support or replace human annotators in SDS development.

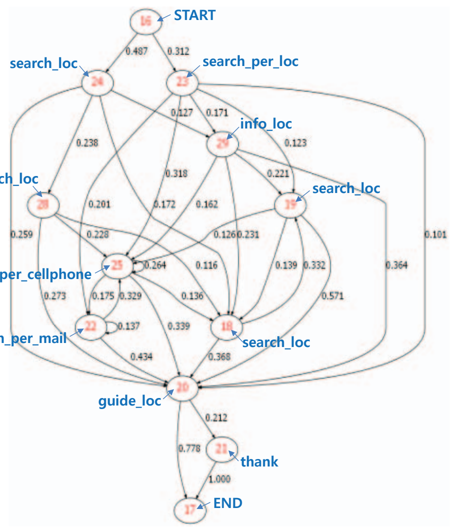

Fig. 4. *An example agenda graph*

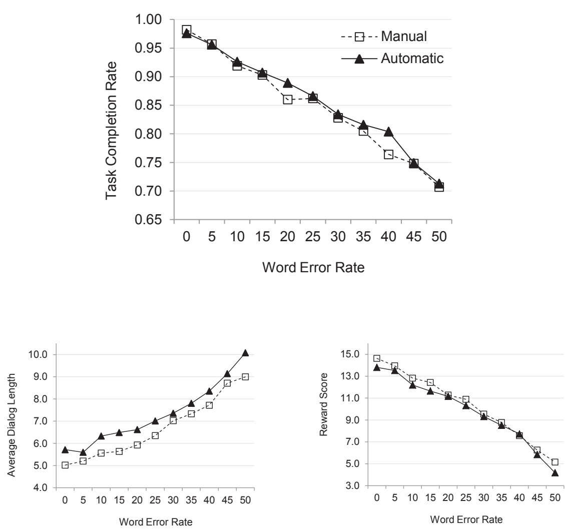

Fig. 5. *Evaluation of dialog systems*

5. CONCLUSIONS

We addressed the problem of unsupervised modeling of user actions in a dialog corpus. We sought to solve the problem using a non-parametric Bayesian Hidden Markov Model (HMM) for unsupervised modeling. One of the main advantages of our model is that a system designer does not need to determine the number of user actions. By adding the word/entity and the background model, the clustering performances are improved. The experimental results for the two dialog systems show that our approach can be applicable to automatically annotate the UAs for the dialog system.

There are several possible subjects for further research on our approach. We can improve the clustering performance by applying additional extensions to the model and by learning the hyper-parameters. We plan to develop an automatic entity annotation method for more rapid development of a dialog system. In addition, we need to evaluate dialog systems in real user environment.

10. ACKNOWLEDGEMENT

This work was supported by the Industrial Strategic technology development program, 10035252, development of dialog-based spontaneous speech interface technology on mobile platform, funded by the Ministry of Knowledge Economy (MKE, Korea).
This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2011-0027953).

11. REFERENCES

[1] J. D. Williams and S. Young, “Partially observable Markov decision processes for spoken dialog systems,” *Computer Speech and Language*, vol. 21, pp. 393-422, April 2007.
[2] C. Lee, S. Jung, S. Kim and G.G. Lee, “Example-based Dialog Modeling for Practical Multi-domain Dialog System,” *Speech Communication*, vol. 51, no. 5, pp. 466-484, May 2009.
[3] G. Tur, D. Hakkani-Tür, and R. E. Schaprie, “Combining active and semi-supervised learning for spoken language understanding,” *Speech Communication*, vol. 45, no. 2, pp. 171-786, February 2005.
[4] M. Jeong, L. Chin-Yew and G.G. Lee, “Semi-supervised speech act recognition in emails and forums,” in *Proc. EMNLP*, 2009, p. 1250-1259.
[5] A. Ritter, C. Cherry, and B. Dolan, “Unsupervised modeling of twitter conversations,” in *Proc. NAACL-HLT*, 2010, p. 172-180.
[6] J. Shafiq, C. Giuseppe and L. Chin-Yew, “Unsupervised modeling of dialog acts in asynchronous conversations,” in *Proc. IJCAI*, 2011.
[7] N. Crook, R. Granell and S. Pulman, “Unsupervised classification of dialogue acts using a Dirichlet process mixture model,” in *Proc. SIGDIAL*, 2009, p. 341-348.
[8] Y. Teh, M. Jordan, M. Beal and D. Blei, “Hierarchical Dirichlet processes,” *Journal of the American Statistical Association*, vol.101, no. 476, pp. 1566-1581, December 2006.
[9] E. Fox, E. Sudderth, M. Jordan and A. Willsky, “An HDP-HMM for systems with state persistence,” in Proc. ICML, 2008, p. 312-319.
[10] S. Goldwater and T. Griffiths, “A fully Bayesian approach to unsupervised part-of-speech tagging,” in *Proc. ACL*, 2007, p. 744-751.
[11] R. Barzilay and L. Lee, “Catching the drift: probabilistic content models, with applications to generation and summarization,” in *Proc. NAACL-HLT*, 2004, p. 113-120.
[12] D. Blei, A. Ng, and M. Jordan, “Latent Dirichlet allocation,” *Journal of Machine Learning Research*, vol. 3, pp. 993-1022, January 2003.
[13] S. Geman and D. Geman, “Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, pp. 721-741, June 1984.
[14] A. Rosenberg and J. Hirschberg, “V-measure: A conditional entropy-based external cluster evaluation measure,” in *Proc. EMNLP-CoNLL*, 2007, p. 410-420.
[15] N. Vinh, J. Epps and J.Bailey, “Information theoretic measures for clusterings comparison: Is a correction for chance necessary?," in *Proc. ICML*, 2009, p. 1073-1080.
[16] C. Lee, S. Jung, K. Kim and G.G. Lee, “Automatic agenda graph construction from human-human dialogs using clustering method,” in *Proc. NAACL-HLT*, 2009, p. 17-20.
[17] S. Jung, C. Lee, K. Kim, M. Jeong and G.G. Lee, “Data-driven user simulation for automated evaluation of spoken dialog systems,” *Computer Speech and Language*, vol. 23, no. 4, pp. 479-509, Oct 2009.
[18] O. Lemon and V. Rieser, “Learning Effective Multimodal Dialogue Strategies from Wizard-of-Oz data:Bootstarpping and Evaluation,” in *Proc. ACL*, 2008, p. 1-1.